

整理番号:

発送番号:630230 発送日:平成20年10月21日

1

引用非特許文献

特許出願の番号

特願2006-521892

作成日

平成20年10月10日

作成者

木村 雅也

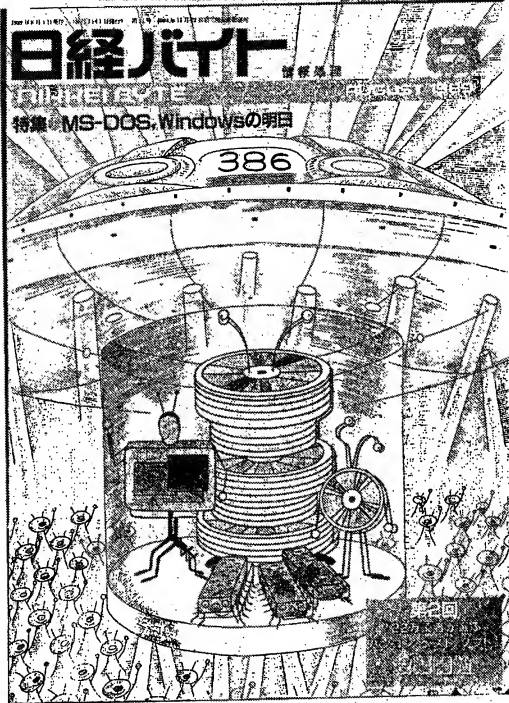
3980 5100

発明の名称

ブランクを移送するためのシステム及び方法



本誌創刊は、特許庁が著作権法第42条第2項第1号の規定により登録したものです。
取扱いにあたっては、著作権侵害とならないよう十分にご注意ください。 発行所: 株式会社 2007-00000-001



本複製物は、特許庁が著作権法第42条第2項第1号の規定により複製したものです。
 取扱いにあたっては、著作権侵害とならないよう十分にご注意ください。 国内技術雑誌2007-00809-005

themes

第4部 高速ファイル

4

EMS メモリとキャッシュの活用で ファイル性能を大幅に改善

フジナ・チェン

- EMS の PC-DOS 4.0 は、ファイル・アクセス性能を大幅に向上させるために、四つの新機構を導入した。
- EMS メモリに最大 1 万個のシステム・バッファを確保できる「ラージ・バッファ」、シーケンシャル・アクセス性能向上のための「ルックアヘッド・バッファ」、開閉に起因するファイル・オープンも高速化する「Fast Open」、検索時に最適化したファイルのランダム・アクセス性能も向上させる「Fast Seek」である。
- EMS の設計者自身が、これらの仕組みを駆使するとともに、性能向上の具体的なデータを示す。(本誌)

PC-DOS は、その登場からまだ 7 年しか経っていないが、最も普及している OS である。IBM PC のハードウェアとアプリケーションの多様化に伴い、DOS も性能を向上してきた。最新のバージョン 4.0 では、ファイル性能を向上させるために四つの機構を実現した (バージョン 3.3 でも一部実現)。

第 1 の機構は、EMS メモリを使った「ラージ・バッファ」だ。大きなファイルのランダム・アクセス性能の向上が目的である。ユーザは最大 1 万個のシステム・バッファを指定できる。

第 2 の機構は、シーケンシャル・ア

ccess のための「ルックアヘッド・バッファ」だ。この目的は、シーケンシャル・アクセスでディスク・アクセス回数を減らすことである。ランダム・アクセス性能を低下させることなく、シーケンシャル・アクセス性能を向上させることができる。ユーザは最大 8 個のルックアヘッド・バッファを指定できる。

第 3 の機構は「Fast Open」だ。これはファイル・ディレクトリ・エントリのキャッシュである。結果にファイルを開閉/クローズする場合に必要

を削減する。

第 4 の機構は「Fast Seek」だ。これはファイル・クラスター領域のキャッシュである。単にディスク上のファイル・クラスター領域をキャッシュするだけでなく、クラスターの連続性など、より密着の深い情報も含む。クラスターを連続したファイルのランダム・アクセスでは、大幅に性能が向上する。

Fast Open と Fast Seek は、ともにキャッシュ内容を LRU 方式¹⁾で管理する。また、メモリをより効率的に使用するように設計した。

フジナ・チェン

著名なフジナ・チェン氏 (Fujina Chen) は、東 RISA 社 (フロッピーディスクのメモリー・インターフェイス・システム) の元開発者で、PC-DOS テクニカル・グループの代表者。PC-DOS テクニカル・グループの代表者。PC-DOS テクニカル・グループの代表者。PC-DOS テクニカル・グループの代表者。

ラージ・バッファ

DOS 4.0 は、Lotus/Intel/Microsoft 社の実用する拡張メモリ仕様 (EMS 4.0) の全機能をサポートする。EMS ドライバ (DOS 4.0 に標準付属) と EMS メモリ・ボードを併用するこ

とで 640 K バイト以上のメモリにアクセスできる。EMS ドライバの設置は、640 K~1 M バイトの間の未使用アドレス空間に実行ページ (10 K バイト単位) をいくつか確保し、これらを EMS

本複製物は、特許庁が著作権法第42条第2項第1号の規定により複製したものです。
取扱いにあたっては、著作権侵害とならないよう十分にご注意ください。

0768-0463/97/0005-0000\$10.00/0

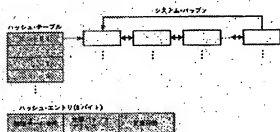


図1 ラージ・バッファの構造。ハッシュ・テーブルはコンテナとスループットメモリ（相互の0と1のビット上の位置関係）に、バッファはRAMメモリに接続される。ハッシュ・テーブルの各エントリは、1個のページ番号、全長（バッファ・アドレス）、送受信履歴、およびそのページ・アドレスである。このうち、送受信履歴はRAMメモリ中のバッファを直接参照する。全長（バッファ・アドレス）は、送受信したデータの総量のバッファ・オフセットとデータ・ワードのセグメント・オフセット・ポイント。各バッファの最大サイズは、ヘッダ領域（バイトとデータ 512 バイトの計 528 バイト）である。受信履歴は更新されたバッファの番号を

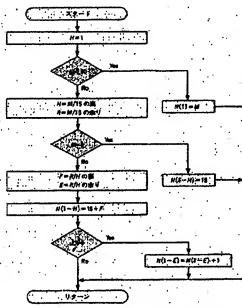


図1 各ハッシュ・エンタリにはばらばらに複数のパッファを管理させるアルゴリズム

(注3) from recently used の略。ヤセラムに空気がない場合、最も近い回アタセスされていない。アタセに前も方式、最近アタセしたデータはヤセラムに記録し可能である。

メモリ内の論理ページ (16 K バイト単位) にマップすることだ。

DOS 4.0では、ファイル・アクセスの高速化のために、一つの物理ページ(物理ページ番号255)を使って、システム・バッファをEMSメモリ中に確保できるようにした。これがラージ・バッファである。

以前のバージョンのDOSと同様に、`compact`、`sys`ファイル中の`buffers`コマンドの出力のパラメータには、ファイル入出力のためのシステム・バッファ数を指定する⁴⁰⁾。ここで新しい/Xオプションを併せて指定すると、最大1万個ものバッファをEMSメモリ中に確保できる。1万個のバッファは、約5MバイトのEMSメモリに相当する。/Xオプションを省略した場合は、バッファは最大99個に制限され、コンプレッションメモリ（従来の540Kバイト以下）の手配に引き渡される。

ハッシングとLRU方式で
ランダム・アクセス性能を向上
ラージ・バッファは、二つの構成条
件に基づいて設計した。

一つは、物理ページの大きさが 16 K バイトであるため、一度にアクセス可能なバッファが 16 K バイトだけであること。

もう一つは、指定されたバッファ取
が何回であっても、バッファを一定時
間内に満満に検索できることだ。

この二つの前提条件を満たすために、
図1に示すようなハッシュ・テーブル
を用いた。

ハッシュ・テーブルの各エントリは、

本複製物は、特許庁が著作権法第42条第2項第1号の規定により複製したものです。
取扱いにあたっては、著作権侵害とならないよう十分にご注意ください。

最新型マイコン搭載

エポニー-II

標準価格 ¥28,000

PCROMの11ヶ所
 プログラムの書き換えが可能
 PCROMの11ヶ所
 プログラムの書き換えが可能

エポニーはS-DOOの
 ファイナルな進化を
 遂げた。

エポニーはS-DOOの
 ファイナルな進化を
 遂げた。

詳しくは下記の資料をご覧ください

MS-DOS2.0
 標準価格 ¥2,000

エコーライトシステム11型電卓がド
 ッキング可能。バックアップ、リ
 ハートアップ、スプレッドシート
 標準価格 ¥1,500

MS-DOS2.0スーパーディスク2型電
 卓がドッキング可能。バックアップ、リ
 ハートアップ、スプレッドシート
 標準価格 ¥1,500

入門用データディスク2型フォーマット
 ディスク 標準価格 ¥1,500

※1. 2. 3. 4. 5. 6. 7. 8. 9. 10. 11. 12. 13. 14. 15. 16. 17. 18. 19. 20. 21. 22. 23. 24. 25. 26. 27. 28. 29. 30. 31. 32. 33. 34. 35. 36. 37. 38. 39. 40. 41. 42. 43. 44. 45. 46. 47. 48. 49. 50. 51. 52. 53. 54. 55. 56. 57. 58. 59. 60. 61. 62. 63. 64. 65. 66. 67. 68. 69. 70. 71. 72. 73. 74. 75. 76. 77. 78. 79. 80. 81. 82. 83. 84. 85. 86. 87. 88. 89. 90. 91. 92. 93. 94. 95. 96. 97. 98. 99. 100. 101. 102. 103. 104. 105. 106. 107. 108. 109. 110. 111. 112. 113. 114. 115. 116. 117. 118. 119. 120. 121. 122. 123. 124. 125. 126. 127. 128. 129. 130. 131. 132. 133. 134. 135. 136. 137. 138. 139. 140. 141. 142. 143. 144. 145. 146. 147. 148. 149. 150. 151. 152. 153. 154. 155. 156. 157. 158. 159. 160. 161. 162. 163. 164. 165. 166. 167. 168. 169. 170. 171. 172. 173. 174. 175. 176. 177. 178. 179. 180. 181. 182. 183. 184. 185. 186. 187. 188. 189. 190. 191. 192. 193. 194. 195. 196. 197. 198. 199. 200. 201. 202. 203. 204. 205. 206. 207. 208. 209. 210. 211. 212. 213. 214. 215. 216. 217. 218. 219. 220. 221. 222. 223. 224. 225. 226. 227. 228. 229. 230. 231. 232. 233. 234. 235. 236. 237. 238. 239. 240. 241. 242. 243. 244. 245. 246. 247. 248. 249. 250. 251. 252. 253. 254. 255. 256. 257. 258. 259. 260. 261. 262. 263. 264. 265. 266. 267. 268. 269. 270. 271. 272. 273. 274. 275. 276. 277. 278. 279. 280. 281. 282. 283. 284. 285. 286. 287. 288. 289. 290. 291. 292. 293. 294. 295. 296. 297. 298. 299. 300. 301. 302. 303. 304. 305. 306. 307. 308. 309. 310. 311. 312. 313. 314. 315. 316. 317. 318. 319. 320. 321. 322. 323. 324. 325. 326. 327. 328. 329. 330. 331. 332. 333. 334. 335. 336. 337. 338. 339. 340. 341. 342. 343. 344. 345. 346. 347. 348. 349. 350. 351. 352. 353. 354. 355. 356. 357. 358. 359. 360. 361. 362. 363. 364. 365. 366. 367. 368. 369. 370. 371. 372. 373. 374. 375. 376. 377. 378. 379. 380. 381. 382. 383. 384. 385. 386. 387. 388. 389. 390. 391. 392. 393. 394. 395. 396. 397. 398. 399. 400. 401. 402. 403. 404. 405. 406. 407. 408. 409. 410. 411. 412. 413. 414. 415. 416. 417. 418. 419. 420. 421. 422. 423. 424. 425. 426. 427. 428. 429. 430. 431. 432. 433. 434. 435. 436. 437. 438. 439. 440. 441. 442. 443. 444. 445. 446. 447. 448. 449. 450. 451. 452. 453. 454. 455. 456. 457. 458. 459. 460. 461. 462. 463. 464. 465. 466. 467. 468. 469. 470. 471. 472. 473. 474. 475. 476. 477. 478. 479. 480. 481. 482. 483. 484. 485. 486. 487. 488. 489. 490. 491. 492. 493. 494. 495. 496. 497. 498. 499. 500. 501. 502. 503. 504. 505. 506. 507. 508. 509. 510. 511. 512. 513. 514. 515. 516. 517. 518. 519. 520. 521. 522. 523. 524. 525. 526. 527. 528. 529. 530. 531. 532. 533. 534. 535. 536. 537. 538. 539. 540. 541. 542. 543. 544. 545. 546. 547. 548. 549. 550. 551. 552. 553. 554. 555. 556. 557. 558. 559. 560. 561. 562. 563. 564. 565. 566. 567. 568. 569. 570. 571. 572. 573. 574. 575. 576. 577. 578. 579. 580. 581. 582. 583. 584. 585. 586. 587. 588. 589. 590. 591. 592. 593. 594. 595. 596. 597. 598. 599. 600. 601. 602. 603. 604. 605. 606. 607. 608. 609. 610. 611. 612. 613. 614. 615. 616. 617. 618. 619. 620. 621. 622. 623. 624. 625. 626. 627. 628. 629. 630. 631. 632. 633. 634. 635. 636. 637. 638. 639. 640. 641. 642. 643. 644. 645. 646. 647. 648. 649. 650. 651. 652. 653. 654. 655. 656. 657. 658. 659. 660. 661. 662. 663. 664. 665. 666. 667. 668. 669. 670. 671. 672. 673. 674. 675. 676. 677. 678. 679. 680. 681. 682. 683. 684. 685. 686. 687. 688. 689. 690. 691. 692. 693. 694. 695. 696. 697. 698. 699. 700. 701. 702. 703. 704. 705. 706. 707. 708. 709. 710. 711. 712. 713. 714. 715. 716. 717. 718. 719. 720. 721. 722. 723. 724. 725. 726. 727. 728. 729. 730. 731. 732. 733. 734. 735. 736. 737. 738. 739. 740. 741. 742. 743. 744. 745. 746. 747. 748. 749. 750. 751. 752. 753. 754. 755. 756. 757. 758. 759. 760. 761. 762. 763. 764. 765. 76

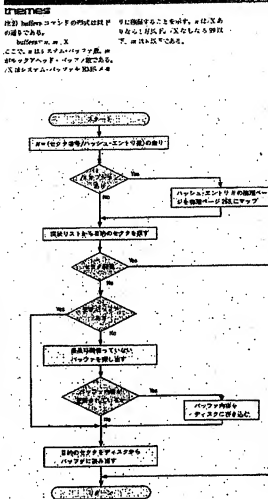


図1 システムバッファのLRUアルゴリズム

押し出して使用する(ただし、バッファ内容が更新されていれば、使用済のバッファ内容をディスクに書き込む)。このアルゴリズムは図3のようになる。

図4はTSP2モデル60で実施した性能テストの結果である。このテスト

では、ハード・ディスク上の250Kバイト・ファイルに対してランダムなセクタ読み書きを1万2800回実行した。バッファ数を増やしていくと、500個あたりから性能がほとんど変わらなくなる。

本誌掲載は、特許庁が著作権法第42条第2項第1号の規定により複製したものです。
取扱いにあたっては、著作権者とならないよう十分に注意ください。

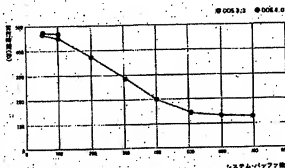


図4 ランダム・アクセスでのランダム・バックアップの信頼性。ハード・ディスク上の2GBのバイト・ソフトに對してランダムなセクタ読み書きを1万1000回実行した。FAT/FAT32で実施した。

これは、ファイルの種とどのセクタがバックアップ内に格納され、ディスクに入出力されてデータにアクセスされるからだ。

ランダム・アクセス・バックアップ

ランダム・アクセス・バックアップの目的は、ランダム・アクセスの間に格納されたセクタを読み出して、ディスク読み出しの回数を減らすことだ。ランダム・アクセス・バックアップの回数は、config.sysファイルのbuffersコマンドの1番目のパラメータで指定する。最大3回、ランダム・アクセス・バックアップを指定できる。

例えば、このパラメータを3とすると、(HDSはコンベンショナル・メモリ中に1個のランダム・アクセス・バックアップを確保する。この状態でディスクからセクタ100を読み出す場合、DOSは先ず100、101、102の3回のセクタを読み出して、次にセクタ101の読み

出しを要求されたとき、ディスク用方なしてランダム・アクセス・バックアップからデータを取り出す。

ランダム・アクセスへの影響はない

DOSは、必要ならランダム・アクセス・バックアップによってランダム読み出しの性能が低下するのを予防する。この予防法は次のような単純なものだ。読み出し要求を受けるたびに、現在要求されているセクタ番号(とすると、以前に要求されたセクタ番号(とすると)を比較する。そして、同じ番号ならセクタを先読みし、そうでなければセクタCだけを読み出す。

図4はFDS/モデル80で実施した

ceanoGRAPHY III

¥22,000

ハードディスクのバックアップ
データ保護のバックアップ
バックアップのバックアップ

最新バックアップ
DMA/IDE/2チャンネル/ハードディスクにより
バックアップ1枚(11MB/127MB)
DOS/バックアップの約8割の
高速復元を実現しています。

8/テリヤ機能
復元時のエラー一掃機能を使用すれば
バックアップ1枚で、2分/1
バックアップ1枚で、2分/1

バックアップ
バックアップ
バックアップ

お求めの価格は、お求めの価格で
お求めの価格で、お求めの価格で
お求めの価格で、お求めの価格で

マイクロテータ
お求めの価格で、お求めの価格で
お求めの価格で、お求めの価格で

本誌掲載は、特許庁が著作権法第42条第2項第1号の規定により複製したものです。
取扱いにあたっては、著作権侵害とならないよう十分にご注意ください。

THEMES

図1 fastopen コマンドの形式は、次のとおりである。

fastopen d [= n]

ここで、d はキャッシュを記憶するディレクトリ・エントリの数。n は Fast Open のキャッシュ・メントリ数。n は Fast Open

ディレクトリ・エントリを含むバッファがフラッシュされる可能性がある。この場合、次の再オープンでは再度、ディスクを読み出さなければならない。

DOS カーネルが利用する 主記憶緩衝型プログラム

こうした非効率を回避するために、Fast Open を設計した。これは、主記憶緩衝型プログラムである。ディレクトリ・エントリを主記憶中にキャッシュすることで、DOS カーネルがディレクトリ・エントリに高速アクセスできるようにする。キャッシュ中のディレクトリ・エントリはプリー構成になっており、LRU 方式で管理する。管理するキャッシュ・エントリ数は、fastopen コマンドの最初のパラメータで決定する。

DOS カーネルと Fast Open 間のインタフェースは、以下のように図1した (図1)。

● Looking: DOS カーネルは、ディスクからディレクトリ・エントリを読み出す際に、いつも Fast Open にルックアップ要求を発行する。この結果は、Found, Partially Found, Unfound の三つのいずれかになる。例えば AYBWCVD というパスを覚えてみよう。まず Found だと、Fast Open は D のディレクトリ・エントリを返す。Partially Found なら、パス中のサブディレクトリ (例えば A, B, または C) のディレクトリ・エントリを返す。Unfound の場合は、DOS カーネルがディスクにアクセスして A, B, C, D のディレクトリ・エントリを検索し

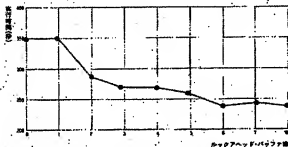


図1 シーケンシャル・アクセスでのルックアップ・バッファの効果。ハードディスク上の256バイト・ファイルに対し、シーケンシャルなセクタ読み書きを1万5000回実行した。256バイト・ファイルで実行した

性能テストの結果である。このテストでは、ハード・ディスク上の256バイト・ファイルに対し、シーケンシャルなセクタ読み書きを1万5000回実行した。また、ランダム読み出し性能低下の予防法の効果を確認するために、

前述のワーク・バッファの性能テストも実施した。その結果、ルックアップ・バッファはシーケンシャル・アクセス性能を大幅に向上させるが、ランダム・アクセス性能には影響しないことがわかった。

Fast Open

ディスク上のファイルは、それぞれ32バイトのディレクトリ・エントリで管理されている。ディレクトリ・エントリは、ファイルの属性や変更日付、時刻、サイズ、ローケーションの始点などの管理情報を含む。ファイルのオープンを要求すると、DOS はそのディレクトリ・エントリをディスクからシステム・バッファに読み出し、その後の読み書きで管理情報を利用して行う。こうした方式を繰り返しているため、同一ファイルに対してオープン、読み出し、書き込み、クローズを頻発に行うアプリケーション・プログラム

では、アクセス効率が悪化することがある。

第1に、32バイトのディレクトリ・エントリを読み出すために、DOS は1セクタ (通常512バイト) をまるごと読み出す必要がある。例えば、インペルの開きのパス名 (A\BWCVD など) は、システム・バッファ中の4K (32768) バイトを無駄にする。その結果、システム・バッファの LRU 方式の効果が悪化する。

第2に、バッファ数に制限があるため、ファイルを開閉した後に実行する読み出しと書き込みによってディ

本複製物は、特許庁が著作権法第42条第2項第1号の規定により複製したものです。
取扱いにあたっては、著作権保護とならないよう十分にご注意ください。

© 1995 Microsoft Corporation
All rights reserved.

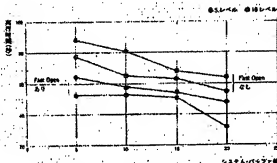


図1 はエントリのFast Openキャッシュの機能。各レベルのサブディレクトリから、他のレベルの深さのサブディレクトリに、ファイルを開くのを10回繰り返した。ここで、10回または15回である。サブディレクトリは100バイト・ファイルを1個持っている。100バイト・ファイルで100回。

または10である。各サブディレクトリは100バイト・ファイルを含んでいる。この結果は、Fast Openのサ

ッシュ・エントリを20個にすると、一貫した性能向上が得られることを示している。

100のディスク・プロセッサの読み込みはクラスターである。これは、一定量のセクタに出現する(クラスター)当たりのセクタ数はフォーマット時に決まる。ユーザの観点からは、ファイルはバイト単位であり、すべてのクラスターは連続的に連続しているように見える。すなわち、ファイルは連続クラスターから始まって連続クラスター、連続クラスターと連続する。しかし、物理的には、ファイルは連続したクラスターで構成されているとは限らない。例えば、連続クラスターが物理クラスター200にマップされたりする。DOSは、ファイルがどのクラスター

を使っているか、未使用のクラスターはどれか、などの情報をファイル・アロケーション・テーブル(FAT)に記録する。FATはディスク上にあり、各FATエントリは各クラスターと1対1に対応している。ファイルの先頭クラスターは、ディレクトリ・エントリが指し示すFATエントリに対応するクラスターである。後続のクラスターはこのFATエントリを次のエントリで指し示すことで管理される。DOSは、ファイルの読み込みの時に、このクラスター・チェーンをたどることで現在のファイル・ポジションを計算するわけだ。

この方法では、Fast Openの場合と異様に非効率になる場合がある。

図1に、FATセクタ(FATを含むセクタ)が複数のファイルのクラスター・チェーンを含むことがある。またクラスター・チェーンが複数のFATセクタにまたがることもある。後者の場合、一つのファイルのクラスター・チェーンをたどるために、DOSはいくつものFATセクタをシステム・バッファに読み出さなければならない。このためにバッファ・スペースが無駄になり、システム・バッファのLRU方式の効率が低下する。

図2に、もし物理クラスターが連続しているなら、迅速となるクラスター番号にオフセットを加算するだけで目的のクラスター番号を計算することができる。この場合でも、クラスター・チェーンを物理的にたどるのとは非効率だ。

クラスターの連続性管理でシーク速度を向上

こうした非効率を軽減するためにFast Seekを設計した。クラスター・チェーンの検索が最速な方法で実行されるように、物理的な連続でクラスター番号をキャッシュするのである。Fast Openと異点が多いので、Fast Seekはfastopenコマンドに一体化してある。Fast Seekのためのキャッシュ・エントリ数は、fastopenコマンドの2番目のパラメータで指定する。

Fast Seekのキャッシュ・エントリは、ファイルのどの部分が物理的に連続しているか、という情報を含んでいる。この情報は連続する物理クラスター群の先頭クラスター番号と最終クラスター番号で表現される。ファイルの連続性が

図100

MS-DOS, Windows 95/98: 英語ファイル名: INDEX.BYTE/AUGUST 1999

本複製物は、特許庁が著作権法第42条第2項第1号の規定により複製したものです。
取扱いにあたっては、著作権侵害とならないよう十分にご注意ください。

地址: 上海南京路中法大藥房對面 電話: 021-23112222

低いほど、キャッシュ・エントリ数は少なくてすむ。複数ファイルのオープンや連続性の低いファイルによってキャッシュ・エントリが使い尽くされた場合、Fast Seek は LRU 方式でキャッシュ・エントリを再使用する。

DOS カーネルと Fast Seek 間のインタフェースは以下のように設計した。

- Open: ファイルがオープンされると、Fast Seek は自動的にファイル・リクエストを作成するか(もし存在しなければ)、グループ・ファイル・リストからファイル・ヘッダを調べる。このファイル・ヘッダが最近使用したものになる。
- Close: ファイルがグループによって閉じると、Fast Seek はファイル・ヘッダとそのキャッシュ・エントリをグループ・ファイル・リストに移す。
- Lookup: DOS のようなものは、ディスク上でファイルのクラス・チェーンをたどる前に、まず Fast Seek にルックアップを実行する。この結果は、Found 又は Partially Found のどちらになる。もし Found だと、後述の Partially Found クラスで返す。
- Partially Found: ファイル・ヘッダが記録されているもののすぐ近い階層クラスで返す場合。結果としては、ファイルの次の階層クラスで書かれた場合だ。
- Insert: ファイル・クラス・チェーンをネスト型に挿入する。ファイルをオープンして初めてアクセスする場合には可能。
- Delete: ファイル・ヘッダとそのキャッシュ・エントリを消去する。ファイルの階層から完全に削除する。



A 4サイズの書類を
電子ライティング

本展覧物は、特許庁が著作権第42条第2項第1号の規定により複製したものであり、
取扱いにあたっては、著作権侵害とならないよう十分にご注意ください。

ATをAXに変える AX-KIT。 プロサイドが '89年に贈る夢!!

同様の性能と
最先端技術の実現

最新のハードウェア
ソフトウェア

P16880DAシリーズ(AX仕様)

P16880AX-8K0D	¥158,000
P16880AX-16K0D	¥168,000
P16880AX-32K0D	¥168,000
P16880AX-64K0D	¥168,000
P16880AX-128K0D	¥168,000
P16880AX-256K0D	¥168,000

P16880DAシリーズ(AT仕様)

P16880-8K0D	¥158,000
P16880-16K0D	¥168,000
P16880-32K0D	¥168,000
P16880-64K0D	¥168,000
P16880-128K0D	¥168,000
P16880-256K0D	¥168,000

従来のハードウェアとソフトウェア
の両方を改良

AX-KITシリーズ

P16880AX-8K0D-Kit	¥168,000
P16880AX-16K0D-Kit	¥168,000
P16880AX-32K0D-Kit	¥168,000
P16880AX-64K0D-Kit	¥168,000
P16880AX-128K0D-Kit	¥168,000
P16880AX-256K0D-Kit	¥168,000

AX-KITシリーズ

P16880AX-8K0D-Kit	¥168,000
P16880AX-16K0D-Kit	¥168,000
P16880AX-32K0D-Kit	¥168,000
P16880AX-64K0D-Kit	¥168,000
P16880AX-128K0D-Kit	¥168,000
P16880AX-256K0D-Kit	¥168,000

AX-KITシリーズ

P16880AX-8K0D-Kit	¥168,000
P16880AX-16K0D-Kit	¥168,000
P16880AX-32K0D-Kit	¥168,000
P16880AX-64K0D-Kit	¥168,000
P16880AX-128K0D-Kit	¥168,000
P16880AX-256K0D-Kit	¥168,000

AX-KITシリーズ

P16880AX-8K0D-Kit	¥168,000
P16880AX-16K0D-Kit	¥168,000
P16880AX-32K0D-Kit	¥168,000
P16880AX-64K0D-Kit	¥168,000
P16880AX-128K0D-Kit	¥168,000
P16880AX-256K0D-Kit	¥168,000

AX-KITシリーズ

P16880AX-8K0D-Kit	¥168,000
P16880AX-16K0D-Kit	¥168,000
P16880AX-32K0D-Kit	¥168,000
P16880AX-64K0D-Kit	¥168,000
P16880AX-128K0D-Kit	¥168,000
P16880AX-256K0D-Kit	¥168,000

AX-KITシリーズ

P16880AX-8K0D-Kit	¥168,000
P16880AX-16K0D-Kit	¥168,000
P16880AX-32K0D-Kit	¥168,000
P16880AX-64K0D-Kit	¥168,000
P16880AX-128K0D-Kit	¥168,000
P16880AX-256K0D-Kit	¥168,000

AX-KITシリーズ

P16880AX-8K0D-Kit	¥168,000
P16880AX-16K0D-Kit	¥168,000
P16880AX-32K0D-Kit	¥168,000
P16880AX-64K0D-Kit	¥168,000
P16880AX-128K0D-Kit	¥168,000
P16880AX-256K0D-Kit	¥168,000

AX-KITシリーズ

P16880AX-8K0D-Kit	¥168,000
P16880AX-16K0D-Kit	¥168,000
P16880AX-32K0D-Kit	¥168,000
P16880AX-64K0D-Kit	¥168,000
P16880AX-128K0D-Kit	¥168,000
P16880AX-256K0D-Kit	¥168,000

AX-KITシリーズ

P16880AX-8K0D-Kit	¥168,000
P16880AX-16K0D-Kit	¥168,000
P16880AX-32K0D-Kit	¥168,000
P16880AX-64K0D-Kit	¥168,000
P16880AX-128K0D-Kit	¥168,000
P16880AX-256K0D-Kit	¥168,000

AX-KITシリーズ

P16880AX-8K0D-Kit	¥168,000
P16880AX-16K0D-Kit	¥168,000
P16880AX-32K0D-Kit	¥168,000
P16880AX-64K0D-Kit	¥168,000
P16880AX-128K0D-Kit	¥168,000
P16880AX-256K0D-Kit	¥168,000

AX-KITシリーズ

P16880AX-8K0D-Kit	¥168,000
P16880AX-16K0D-Kit	¥168,000
P16880AX-32K0D-Kit	¥168,000
P16880AX-64K0D-Kit	¥168,000
P16880AX-128K0D-Kit	¥168,000
P16880AX-256K0D-Kit	¥168,000

AX-KITシリーズ

P16880AX-8K0D-Kit	¥168,000
P16880AX-16K0D-Kit	¥168,000
P16880AX-32K0D-Kit	¥168,000
P16880AX-64K0D-Kit	¥168,000
P16880AX-128K0D-Kit	¥168,000
P16880AX-256K0D-Kit	¥168,000

AX-KITシリーズ

P16880AX-8K0D-Kit	¥168,000
P16880AX-16K0D-Kit	¥168,000
P16880AX-32K0D-Kit	¥168,000
P16880AX-64K0D-Kit	¥168,000
P16880AX-128K0D-Kit	¥168,000
P16880AX-256K0D-Kit	¥168,000

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

AX-KITシリーズは、最新のハードウェアとソフトウェアの両方を改良した製品です。

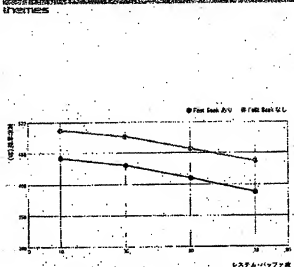


図1. FSCのFast Seekキャッシュの効果。テスト内容は図4と同じランダム・アクセス

● Truncate: キャッシュからファイル

のクスタデータを切り捨てる。ファイ

ルを小さくした場合などに使う。

● Purge: キャッシュ内のすべてのク

スタデータを消去する。ディスクのフ

ォーマットや、絶対セクタの書き込み

要求 (fast seek) があつた場合などに使

用する。

図1はFast Seek性能のテスト結

果である。テスト方法は、前述のラン

ジ・マップの場合と同一である。こ

の結果は、10回のキャッシュ・エン

トリを測定すると、一貫して約40秒毎

であることを示している。システム・

マップが10回刻のとき、Fast Seekを

使わずに同程度の性能(440秒)を得る

ために、さらに60回刻のマップが必

要だ。これは30Kバイトのメモリ

に相当するが、Fast Seekを使うなら

5.4Kバイトのメモリで済む。

こうした結果は、すべてのファイル

・クスタデータをセーブするための十

分なキャッシュ・エントリを必要とし

一般に期待できる。最悪の場合は、フ

ォーマット(大抵かつ非連続)であり、ディ

スクにまばらに分散配置されている

場合だ。この場合、ランダム・アクセ

スを低減に行くと、LRU検索が有効に

使えず、Fast Seekは有効な対策を見

えなくなる。こうした問題を解決する

方法の一つは、キャッシュ・エントリ

の活用がなくなるまで、Fast Seek

を一時的にシャットダウンすること

である。

● ● ●

以上、DOSの二つの新しい機構を説

明した。これのおかげで、config.sys

ファイルのbuffersコマンドや、file

topenコマンドを無視し、アプリケーション性能を向上させることが

できるようになった。同じアプリケーション・プログラムの同一コンピュ

タで実行した場合でも、ファイル・

アクセス速度が向上したことに気づく

だろう。

(BYTE, c. 1989 McGraw-Hill, Inc.)

● ● ●

以上、DOSの二つの新しい機構を説

明した。これのおかげで、config.sys

ファイルのbuffersコマンドや、file

topenコマンドを無視し、アプリケーション性能を向上させることが

できるようになった。同じアプリケーション・プログラムの同一コンピュ

タで実行した場合でも、ファイル・

アクセス速度が向上したことに気づく

だろう。

(BYTE, c. 1989 McGraw-Hill, Inc.)

● ● ●

以上、DOSの二つの新しい機構を説

明した。これのおかげで、config.sys

ファイルのbuffersコマンドや、file

topenコマンドを無視し、アプリケーション性能を向上させることが

できるようになった。同じアプリケーション・プログラムの同一コンピュ

タで実行した場合でも、ファイル・

アクセス速度が向上したことに気づく

だろう。

(BYTE, c. 1989 McGraw-Hill, Inc.)

● ● ●

以上、DOSの二つの新しい機構を説

明した。これのおかげで、config.sys

ファイルのbuffersコマンドや、file

topenコマンドを無視し、アプリケーション性能を向上させることが

できるようになった。同じアプリケーション・プログラムの同一コンピュ